

BAB I

PENDAHULUAN

1.1 Latar Belakang

Server cluster merupakan terdiri dari n *server* yang terhubung melalui jaringan berkecepatan tinggi untuk menyelesaikan permintaan pengguna. Setiap *server* bertindak sebagai *node* yang memiliki *disk* dan sistem operasi sendiri dan *load balancing* yang sederhana dan terdistribusi. Meskipun *cluster* memiliki banyak *server*, namun hanya menggunakan satu nama *host* dan IP *virtual* untuk menyediakan *interface* bagi pengguna (Xu & Wang, 2014).

Server cluster melakukan pendistribusian beban kerja (*load balancing*) kepada sejumlah n *server* yang berada didalam *server cluster* melalui sebuah *load balancer*. Salah satu algoritma *Load Balancing* adalah *least-connection*. Algoritma ini bekerja dengan mendistribusikan *request* dari *client* ke *cluster* berdasarkan jumlah koneksi. Algoritma ini digunakan karena mendistribusikan trafik secara adil (Mustafa, 2017).

Permasalahan *server cluster* adalah penggunaan daya listrik. *Data center*, sebagai tempat menyimpan *server cluster*, di Indonesia mengkonsumsi 1.5% kapasitas pembangkit nasional pada tahun 2014. Presentase ini meningkat pada tahun 2017 menjadi 2% kapasitas pembangkit nasional atau setara dengan konsumsi kapasitas pembangkit untuk gabungan Jambi, Riau dan Sumatera Barat (Asian Development Bank, 2017). Salah satu cara menanggulangi masalah penggunaan daya listrik adalah memadamkan *node server* pada *server cluster* (Orgerie, Assuncao, & Lefevre, 2014). Pemadaman *node server* memberikan pengaruh terhadap penggunaan arus listrik. Jika seluruh *node* pada *server cluster* dinyalakan tanpa dibebani *load*, arus listrik yang digunakan adalah sebesar 0.9A. Jika salah satu *node* dipadamkan, arus listrik yang digunakan berkurang menjadi 0.7A (Harja, 2016).

Pemadaman *node server* tidak boleh dilakukan tanpa perencanaan karena akan mempengaruhi kualitas *server*. Oleh karena itu diperlukan penjadwalan pemadaman *node server* yang efektif dengan mempertimbangkan kinerja *node*

server. Salah satu indikator kinerja *node server* adalah *load average*. *Load average* mengindikasikan beban kerja yang sedang diproses oleh CPU (Walker, 2006). *Load average* adalah nilai rata-rata dari jumlah proses yang sedang berstatus *runnable* atau *uninterruptable*. Nilai *load average* bergantung pada jumlah *core*. Nilai *load* 1.00 pada *single core* berarti penuh sedangkan pada *dual core* hanya 50% (Andre, 2009).

Dengan beragamnya jumlah *core* pada *server*, diperlukan *clustering*. *Clustering* merupakan suatu teknik analisis multivariat yang bertujuan untuk mengelompokkan data observasi ataupun variabel-variabel ke dalam *cluster* sedemikian rupa sehingga masing-masing *cluster* bersifat homogen sesuai dengan faktor yang digunakan untuk melakukan pengelompokan (Gundono, 2011).

Terdapat banyak algoritma *clustering*, salah satunya adalah gabungan algoritma *Hierarchical clustering* dan K-Means *clustering*. Gabungan algoritma *Hierarchical clustering* dan K-Means *clustering* dapat menghasilkan pengelompokan data yang lebih baik bila dibandingkan dengan K-Means *clustering*. Metode *average linkage hierarchical clustering* dipilih karena menghasilkan nilai koefisien korelasi *cophenetic* (Alfina, Santosa, & Barakbah, 2012).

Pada penelitian ini *server cluster* akan menggunakan metode *load balancing least connection*. Saat *server cluster* berjalan dilakukan pengambilan data *load average* yang dibutuhkan sebagai parameter penjadwalan. Parameter yang dibutuhkan adalah *load average* satu, lima dan 15 menit. Pengambilan data dilakukan setiap menit. Kemudian data yang terkumpul diklasterisasi menggunakan gabungan algoritma *average linkage hierarchical clustering* dan K-Means. Hasil dari klasterisasi tersebut menghasilkan tiga *cluster load average* yaitu “rendah”, “sedang” dan “tinggi”. *Load average* yang termasuk ke dalam kategori “rendah” diurutkan berdasarkan waktu pengambilan data sehingga mendapatkan waktu dan durasi pemadaman pada setiap *node*.

Dengan dipadamkannya *node server* berpengaruh terhadap kualitas *server* dan penggunaan daya listrik. Untuk itu perlu dilakukan pengujian kualitas *server* dan penggunaan daya listrik sebelum dan sesudah dilakukan pemadaman

berdasarkan hasil dari pengolahan data *uptime*. Pengujian kualitas *server* menggunakan parameter *availability*, *throughput* dan *packet loss*. *Availability* mengacu pada sistem yang dapat memberikan layanan pada kondisi operasi normal selama penggunaan aplikasi untuk pertukaran data. *Availability* menunjukkan persentase durasi operasi normal sistem (Lyu, Li, Yan, Qian, & Sheng, 2017). *Throughput* adalah tingkat penerimaan data yang sukses dalam interval waktu yang diukur dalam bit/detik (Kurose & Ross, 2008). *Packet loss* adalah kondisi dimana paket telah dikirim namun tidak pernah mencapai tempat tujuan (Kurose & Ross, 2008).

Dengan diadakannya penjadwalan pemadaman *node server* pada *server cluster* diharapkan dapat menentukan waktu dan durasi pemadaman sehingga dapat mengefisiensi penggunaan daya listrik dan menjaga kualitas *server* secara maksimal.

1.2 Rumusan Masalah

Rumusan masalah dalam penelitian ini adalah:

1. Apakah sistem untuk mengatur penjadwalan pemadaman *node server* pada *server cluster* berhasil dibangun?
2. Apakah penjadwalan pemadaman mampu memberikan waktu dan durasi pemadaman pada *node server* pada *server cluster*?
3. Apakah penggunaan penjadwalan pemadaman *node server* berhasil mengurangi penggunaan daya dan tetap menjaga kualitas *server cluster*?

1.3 Tujuan Penelitian

Sejalan dengan permasalahan yang telah dirumuskan, maka tujuan yang ingin dicapai pada penelitian ini adalah:

1. Membangun sistem untuk mengatur penjadwalan pemadaman *node server* pada *server cluster*.
2. Untuk memberikan waktu dan durasi pemadaman *node server* pada *server cluster*.

3. Membuktikan penggunaan penjadwalan pemadaman *node server* pada *server cluster* mampu mengurangi daya listrik dengan tetap menjaga kualitas *server*.

1.4 Batasan Masalah

Batasan masalah yang diteliti antara lain adalah :

1. Metode klasterisasi yang dipakai dalam proses *data clustering* adalah gabungan algoritma *Average Linkage Hierarchical* dan *K-Means Clustering*,
2. Algoritma yang digunakan untuk *load balancing* adalah *least connection*.
3. Tingkat kesibukan *server* diukur dengan menggunakan *load average* dalam satu menit, lima dan 15 menit,
4. Parameter pengujian kualitas *server* yang digunakan adalah *availability*, *throughput* dan *packet loss*.

1.5 Manfaat Penelitian

Adapun manfaat dari penelitian ini adalah perangkat lunak yang dapat membantu *system administrator* dalam mengatur pemadaman *node* dari *server* yang mereka kelola dengan durasi yang sesuai dan tetap menjaga `kualitas *server*.

1.6 Sistematika Penulisan

Dalam penyusunan skripsi ini, sistematika penulisan dibagi menjadi beberapa bab sebagai berikut:

BAB I PENDAHULUAN

Bab ini merupakan gambaran umum yang melatarbelakangi penelitian ini yaitu, penggunaan daya listrik pada *server cluster* yang besar. Penelitian ini bertujuan untuk membangun sistem penjadwalan pemadaman *node server* pada *server cluster* yang diharapkan dapat menentukan waktu dan durasi pemadaman sehingga dapat mengefisiensi penggunaan daya listrik dan menjaga kualitas *server* secara maksimal dengan beberapa batasan yang perlu diperhatikan. Manfaat penelitian ini adalah membantu *system administrator* dalam mengatur pemadaman *node* dari *server* yang mereka kelola.

BAB II KAJIAN PUSTAKA

Bab ini memaparkan teori dan konsep yang mendukung dalam penelitian seperti penelitian terdahulu, *server cluster*, *load balancing*, *least connection*, *data clustering*, *hierarchical clustering*, *average linkage*, *K-Means clustering*, *load average*, *availability*, *throughput*, *packet loss*, NGINX dan Siege.

BAB III METODOLOGI PENELITIAN

Bab ini merupakan menjelaskan bagaimana langkah-langkah yang dilakukan pada penelitian ini, langkah-langkah penelitian meliputi tahapan awal penelitian, pengembangan *server cluster* yang mencakup topologi dan konfigurasi, pengembangan perangkat lunak yang mencakup analisis kebutuhan, *input*, *output* dan rancangan perangkat lunak serta bagaimana pengujian daya dan kualitas *server* dilakukan.

BAB IV HASIL PENELITIAN DAN PEMBAHASAN

Pada bab ini membahas bagaimana hasil implementasi *server cluster* yang sudah rancang, hasil implementasi perangkat lunak yang sudah dikembangkan meliputi *input* yang dibutuhkan, setiap proses yang dilakukan dan *output* yang dihasilkan oleh perangkat lunak serta membahas hasil pengujian daya listrik dan kualitas *server*.

BAB V KESIMPULAN DAN SARAN

Pada bab ini menjelaskan kesimpulan yang didapatkan yang merupakan jawaban dari rumusan masalah penelitian, serta berisi saran yang dapat menjadi rujukan untuk penelitian selanjutnya.